УДК 004.031.4

І. Баран, О. Дуда, О. Маєвський
(Тернопільський національний технічний університет імені Івана Пулюя)

РОЗШИРЕННЯ ФУНКЦІОНАЛЬНИХ МОЖЛИВОСТЕЙ PHP ДЛЯ ПЕРЕВІРКИ ОТРИМАНИХ ВІД КОРИСТУВАЧА ДАНИХ

Одним з важливих питань для веб-програмістів є безпека PHP-скриптів. Більшість вразливостей PHP-скриптів є наслідком недостатньої фільтрації отриманих від користувача даних. Здебільшого використовуються декілька прийомів фільтрації з використанням функцій addslashes, mysql_escape_string, mysql_real_escape_string та htmlspecialchars. Використання цих функцій не враховує кодування БД і допускає можливості обходу фільтрації [1]. Для захисту веб-сайтів доцільно розробити уніфіковані функції, які перевірятимуть всі отримані від користувача значення на відповідність типу даних та шаблонам регулярних виразів з врахуванням кодування.

Розглянемо послідовність реалізації подібних функцій. При ініціалізації функції задається вхідний текст `$inp_text`, кодування `$enc` (наприклад `"utf-8"`, `"windows-1251"`, іт.п.) та тип результату. В якості типу результату можна вказувати наперед задані ключові слова (наприклад `"name"` – для імені користувача). Після ініціалізації відбувається вибір шаблону регулярного виразу для заданого типу даних:

```
function check_data($inp_text,$enc,$data_type){
      switch($data_type){
case"name":$f_pattern="/[a-яA-ЯЁёiIїЇєЄрРтТьЬшШщЩуУхХъЪыЫэЭюЮ\-
]{1,};
            $break;
            ...
case"email":$f_pattern="/^[a-zA-Z0-9_\-.]+@[a-zA-Z0-9\-]+\.[a-zA-Z0-
9\-.]+/";break;
            default: $f_pattern=0;}
```

Якщо вибране значення шаблону та кодування отриманих від користувача даних `$enc` відрізняється від базового кодування скрипта, відбувається перекодування значення `$f_pattern`.

```
      if($f_pattern){
            if($enc!="utf-8"){$f_pattern=iconv("utf-
      8",$enc,$f_pattern);}
            $inp_text=htmlspecialchars(stripslashes($f_inp));

            if(preg_match($f_pattern,$inp_text,$f_filterd)){
                  $returned= $f_filterd[0];
            else{$returned=0;}
      else{$returned=0;}
      return($returned); }
```

Наступним кроком відбувається використання перетворення спеціальних символів в HTML-сутності та видалення екранування символів за допомогою функцій `htmlspecialchars` та `stripslashes` відповідно. На завершення виконується перевірка на відповідність шаблону та функція повертає отримане значення `$returned`.

1. Фильтрация и проверка данных PHP. Частые ошибки – Режим доступу: http://habrahabr.ru/post/143035/ – Назва з екрану. – Дата звернення: 30.04.2014.